AMENDMENTS TO THE CLAIMS

Please add or amend the claims to read as follows and cancel without prejudice to filing in a divisional or continuation application the claims marked as cancelled:

1. (Currently Amended) A method comprising:

generating branch predictions for two sequential lines in parallel during a prediction ~~period;~~ by:

determining if data stored in entries of the first side or the second side of a cache of a branch predictor indicates that a branch is to be taken by either of the entries; the cache being segmented into the first side and the second side, where the entries on the first side correspond to a set of addresses having even-numbered indexes, and entries on the second side correspond to a set of addresses having odd-numbered indexes;

storing said branch predictions in a queue; and

delivering a stored branch prediction from said queue to an instruction fetch unit.

2. (Previously Presented) The method as in claim 1, wherein said prediction period comprises two clock cycles.

3. (Cancelled)

4. (Currently Amended) The method as in claim [[3]] 1, wherein an index of one of two sequential lines corresponds to an entry on said first side of said cache, and an index of another of said two sequential lines corresponds to an entry on said second side of said cache.

5. (Cancelled)

6. (Original) The method as in claim 1, comprising generating branch predictions for a stream of addresses during a stall of said instruction fetch unit.

7. (Original) The method as in claim 1, comprising generating during a cycle a prediction for a line, said line being other than the line being fetched by said instruction fetch unit during said cycle.

8. (Original) The method as in claim 1, wherein the addresses for which predictions are generated by a branch prediction unit are decoupled from the addresses for which lines are fetched by said instruction fetch unit.

9. (Cancelled)

10. (Cancelled)

11. (Previously Presented) The method as in claim 1, comprising delivering a branch prediction to said instruction fetch unit in the same prediction period as said branch prediction is written to said queue.

12. (Currently Amended) A processor comprising:

an instruction fetch unit; and

a branch prediction unit, said branch prediction unit comprising a queue to store branch predictions, said branch prediction unit to generate branch predictions for two sequential lines in parallel during a prediction period, and said branch prediction unit to deliver branch predictions stored in said queue to said instruction fetch ~~unit.~~ unit, wherein the branch prediction unit comprises a cache whose entries are segmented into a first side and a second side, where entries on the first side correspond to a set of addresses having even-numbered indexes, and entries on the second side correspond to a set of addresses having odd-numbered indexes.

13. (Cancelled)

14. (Currently Amended) The processor as in claim ~~[[13]]~~ 12, wherein said cache is to store odd-numbered addresses in a first segment of said cache, and even-numbered addresses in a second segment of said cache.

15. (Previously Presented) The processor as in claim 12, wherein said prediction period comprises two clock cycles.

16. (Original) The processor as in claim 12, wherein said branch prediction unit is, in a prediction period, to write a branch prediction to said queue and to deliver said branch prediction to said instruction fetch unit.

17. -19. (Cancelled)

20. (Currently Amended) A system comprising:

a dynamic random access memory unit; and

a processor comprising:

an instruction fetch unit; and

a branch prediction unit, said branch prediction unit comprising a queue to store branch predictions, said branch prediction unit to generate branch predictions for two sequential lines in parallel during a prediction period, and said branch prediction unit to deliver branch predictions stored in said queue to said instruction fetch ~~unit.~~ unit, wherein

the branch prediction unit comprises a cache whose entries are segmented into a first side and a second side, where entries on the first side correspond to a set of addresses having even-numbered indexes, and entries on the second side correspond to a set of addresses having odd-numbered indexes.

21. (Cancelled)

22. (Currently Amended) A system as in claim [[21]] 20, wherein said cache is configured to store odd addresses in said odd side, and even addresses in said even side.

23. (Previously Presented) The system as in claim 20, wherein said prediction period comprises two clock cycles.

24. (Previously Presented) The system as in claim 20, wherein said branch prediction unit is, in a prediction period, to write a branch prediction to said queue and to deliver said branch prediction to said instruction fetch unit.

25. (Previously Presented) The system as in claim 20, wherein the addresses for which predictions are generated by said branch prediction unit are decoupled from the addresses for which lines are fetched by said instruction fetch unit.